Remarks

An amendment presenting rejected claims in better form for consideration on appeal may be admitted. *See* 37 C.F.R. § 116(b)(2). Accordingly, reconsideration of this Application is respectfully requested.

Upon entry of the foregoing amendment, claims 8-12, 14-18 and 20-42 are pending in the application, with claims 8, 14, 30, 36, 38 and 41 being the independent claims. Claims 1-7, 13 and 19 were previously canceled. Claim 38 is sought to be amended. This change introduces no new matter and its entry is respectfully requested.

Based on the foregoing amendments and the following remarks, Applicants respectfully request that the Examiner reconsider all outstanding objections and rejections and that they be withdrawn.

Personal Interview

Applicants thank the Examiner for the personal interview conducted on May 11, 2007. During the personal interview, Applicants' representatives and the Examiner discussed independent claims 8, 14, 30, 36, 38, and 41 and the applied references. An agreement was not reached. The arguments made during the personal interview are incorporated by reference and expanded herein.

Allowable Claims

Applicants thank the Examiner for indicating that claims 20-24 are allowable over the cited references.

Rejections under 35 U.S.C. § 103

Claims 8-12, 14-18, and 25-29

Claims 8-12, 14-18 and 25-29 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 5,488,729 to Vegesna *et al.* ("Vegesna") in view of U.S. Patent No. 5,481,734 to Yoshida ("Yoshida"). Applicants respectfully traverse.

As previously presented, independent claim 8 recites a superscalar microprocessor for processing instructions. The superscalar microprocessor comprises:

> an instruction fetch unit configured to fetch instructions from an instruction store according to a sequential program order;
>
> a branch prediction circuit configured to provide a branch bias signal indicating whether a conditional branch controlled by a conditional branch instruction is predicted to be taken or not taken;
>
> an instruction buffer coupled to receive fetched instructions from the instruction fetch unit and configured to buffer a plurality of fetched instructions, including an instruction selected according to the branch bias signal;
>
> a plurality of functional units configured to execute instructions, thereby generating result data;
>
> a register file including a plurality of entries configured to store data including result data generated by the plurality of functional units, wherein each of the plurality of entries is accessible by reference to a respective location in the register file;
>
> a resource identifying circuit configured to concurrently identify execution resources for a plurality of buffered instructions, thereby making a plurality of instructions concurrently available for issue, wherein the identified execution resources for each of the available instructions includes a functional unit capable of executing the instruction;
>
> a register rename circuit configured to provide references to locations in the register file for logical register references included with the plurality of buffered instructions;
>
> an issue control circuit coupled to the resource identifying circuit and configured to concurrently issue more than one of the available instructions to the functional units for execution, based on availability of the identified execution resources for each instruction and availability of respective operands for each instruction in the

> referenced locations in the register file, without regard to the sequential program order;
>
> a plurality of data routing paths coupled between the plurality of functional units and the register file and configured to concurrently transfer result data from more than one of the plurality of functional units to the register file; and
>
> bypass control logic coupled to the plurality of data routing paths and configured to distribute result data from a first one of the plurality of functional units as operand data for another one or more of the plurality of functional units via an alternate data path that bypasses the register file, wherein distributing result data via the alternate data path occurs concurrently with transferring result data to the register file.

Vegesna and Yoshida, alone or in combination, do not teach or suggest each and every feature of independent claim 8. For example, neither Vegesna nor Yoshida teach or suggest "a branch prediction circuit configured to provide a branch bias signal indicating whether a conditional branch controlled by a conditional branch instruction is predicted to be taken or not taken," as in claim 8. The Examiner has not asserted that Yoshida teaches or suggests this claim feature. Applicants agree.

Vegesna also fails to teach or suggest this claim feature. Vegesna describes a CPU architecture with symmetric scheduling to achieve multiple instruction launch and execution. *See* Vegesna at the Title. According to Vegesna's architecture, a branch instruction is handled by testing the condition associated with the branch instruction during the decode stage of the processor pipeline and then fetching a target instruction based on the actual results of the condition:

> As previously discussed, the branch is decoded and executed during the (D) pipestage. The PCU (6, FIG. 18) calculates the effective address of the target using the address of the branch instruction itself and the offset value specified in the imm22 bit field of the instruction. . . . The condition codes are evaluated by the PCU (6, FIG. 19) and the delay slot instruction [dsi] fetch is also initiated during (D). During the (E) pipestage, the target instruction is fetched if the branch is to be taken, otherwise the next sequential instruction after the dsi is fetched.

Vegesna, col. 25, ll. 55-67. Thus, Vegesna does not teach the *prediction* of whether a branch is taken or not. Rather, Vegesna teaches the *execution* of a branch instruction to determine whether the branch is to be taken or not. Accordingly, Vegesna does ***not*** teach or suggest a ***branch prediction circuit***, let alone a branch bias signal, as recited in independent claim 8.

As far as Applicants understand the Examiner's rejection of claim 8, the Examiner asserts that Vegesna teaches a branch prediction circuit because a signal must be used to differentiate between a state in which the branch has been taken and a state in which the branch has not been taken. (*See* the Office Action at p. 3, ¶ 8.) This assertion, however, misapprehends the plain language of claim 8. In claim 8, the branch bias signal indicates whether a conditional branch controlled by a conditional branch instruction is ***predicted*** to be taken or not taken — *i.e.*, ***before*** the branch instruction is executed. In stark contrast, Vegesna may provide a signal indicating ***that*** a branch has been taken or not — *i.e.*, ***after*** the branch instruction has been executed. Thus, Vegesna does not teach or suggest a branch prediction circuit as in claim 8.

During the personal interview, the Examiner further asserted that Vegesna teaches a branch prediction circuit as claimed, because Vegesna discloses: (i) a first type of design scheme that handles a branch instruction by always assuming that the branch will be taken; and (ii) a second type of design scheme that handles a branch instruction by always assuming the branch will not be taken. *See* Vegesna col. 10, l. 66 - col. 11, l. 25. According to the Examiner, the first type of design scheme is a first bias mode and the second type of design scheme is a second bias mode. Therefore, the Examiner

concludes that Vegesna discloses the branch prediction circuit as claimed. This conclusion, however, neglects language clearly recited in independent claim 8.

Claim 8 includes a branch prediction circuit that enables the microprocessor to handle conditional branch instructions differently depending on whether a branch associated with a conditional branch instruction is predicted to be taken or not. For example, the Specification of the instant application describes two different example methods for processing a conditional branch instruction depending on whether the branch bias signal indicates that the branch is predicted to be taken (*see* the Specification at ¶¶ [0120]-[0134]) or not taken (*see* the Specification at ¶¶ [0135]-[0149]). Thus, the branch prediction ***circuit*** of independent claim 8 enables the microprocessor to ***dynamically*** handle conditional branch instructions in an efficient manner.

In contrast, the microprocessor disclosed by Vegesna does ***not*** include a branch prediction circuit enabling dynamical handling of conditional branch instructions. Rather, Vegesna's architecture is described in terms of the SPARC instruction set. *See* Vegesna, col. 17, ll. 61-64. This architecture handles branch instructions by trivially assuming that a branch will not be taken, which creates a one cycle delay. *See* Vegesna, col. 11, ll. 29-31. To fill the one cycle delay, Vegesna teaches the use of a delay-slot instruction. *See* Vegesna, col. 11, ll. 26-38; col. 25, 62-67. Vegesna's architecture does not permit dynamical handling of conditional branch instructions by providing a branch bias signal that indicates whether a branch will be taken or not, as in independent claim 8. Thus, Vegesna does not render obvious a branch prediction circuit as in independent claim 8.

Furthermore, Vegesna tends to teach away from a branch prediction circuit as in independent claim 8, notwithstanding Vegesna disclosure of a first design scheme that always assumes a branch will be taken (*see* Vegesna, col. 11, ll. 16-25) and a second design scheme that always assumes a branch will not be taken (*see* Vegesna, col. 10, l. 66 - col. 11, l. 15). Vegesna's architecture executes instructions in-order (*see* Vegesna, col. 22, ll. 47-49), fetches one instruction per cycle (*see* Vegesna, col. 17, ll. 63-64), and uses branch delay slots to fill a delay caused by a branch instruction (*see* Vegesna, col. 11, ll. 26-38; FIG. 23). As a result, there is no performance loss caused by Vegesna's trivial "not-taken" branch prediction design scheme, because a one cycle delay caused by a branch instruction is filled by a delay-slot instruction. *See* Vegesna at FIG. 23. Thus, there is no incentive for Vegesna to include a branch prediction circuit as in independent claim 8, because such a branch prediction circuit would merely add unnecessary complexity to Vegesna's architecture without improving performance. Accordingly, Vegesna does not even *suggest* a branch prediction circuit as in independent claim 8.

Because Vegesna and Yoshida, alone or in combination, do not teach or suggest each and every feature of independent claim 8, under the Graham factors, no prima facie rejection of claim 8 has been met. Dependent claims 9-12, 25 and 26 are likewise patentable over the combination of Vegesna and Yoshida for at least the same reasons as independent claim 8 from which they depend, and further in view of their own respective features. Accordingly, Applicants respectfully request that the rejection of claims 8-12, 25 and 26 under 35 U.S.C. § 103(a) be reconsidered and withdrawn.

Independent claim 14 is directed to a method for processing instructions in a superscalar microprocessor. Among other features, the method of claim 14 includes

"predicting whether a conditional branch controlled by a conditional branch instruction included in the fetched instructions is taken or not taken." For at least the same reasons as set forth above with respect to claim 8, neither Vegesna nor Yoshida teach or suggest this feature. Thus, independent claim 14 is patentable over the combination of Vegesna and Yoshida. Dependent claims 15-18 and 27-29 are likewise patentable over the combination of Vegesna and Yoshida for at least the same reasons as independent claim 14 from which they depend, and further in view of their own respective features. Accordingly, Applicants respectfully request that the rejection of claims 14-18 and 27-29 under 35 U.S.C. § 103(a) be reconsidered and withdrawn.

Claims 30-42

Claims 30-42 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 4,942,525 to Shintani *et al.* ("Shintani") in view of U.S. Patent No. 4,594,655 to Hao *et al.* ("Hao"). Applicants respectfully traverse.

As previously presented, independent claim 30 recites a method of executing instructions using a microprocessor, the method comprising:

> pre-fetching an instruction group including a plurality of instructions from a memory in a first processor cycle and holding the instruction group in a pre-fetch buffer, the pre-fetching accomplished so that instruction groups can be returned out of program order and subsequently reordered;
>
> transferring the instruction group held in the pre-fetch buffer to a multiple-stage buffer when there is a vacancy in the multiple-stage buffer with sufficient capacity to handle the instruction group as a unit;
>
> simultaneously decoding, in a processor cycle after the first processor cycle, a plurality of instructions that are included in the instruction group, the decoding performed with at least one instruction at a predetermined position in multiple-stage buffer;
>
> checking, in a processor cycle after the decoding, a dependency relation between plural decoded instructions at least on the basis of registers to be used by the plural decoded instructions;

> allocating instructions, for instructions judged to have no restriction on execution due to dependency, to a plurality of functional units so that at least one instruction executes outside the program order;
>
> executing, in a processor cycle after decoding, the allocated instructions using the plurality of functional units; and
>
> removing the instruction group held in the multiple-stage buffer from the buffer and advancing an other instruction group held in the multiple-stage buffer so that another instruction from the other instruction group moves to the predetermined position.

Shintani and Hao, alone or in combination, do not teach or suggest each and every feature of independent claim 30. For example, neither Shintani nor Hao teach or suggest "transferring the instruction group held in the pre-fetch buffer to a multiple-stage buffer when there is a vacancy in the multiple-stage buffer with sufficient capacity to handle the instruction group as a unit" and "removing the instruction group held in the multiple-stage buffer from the buffer and advancing an other instruction group held in the multiple-stage buffer so that another instruction from the other instruction group moves to the predetermined position," as recited in independent claim 30. The Examiner has not asserted any reason why Hao renders these features unpatentable. Applicants agree that no such reasons exist.

Shintani also does not teach or suggest these claim features. Shintani is directed to a data processor for concurrent execution of instructions by plural execution units. *See* Shintani at the Title. Although Shintani appears to teach pre-fetching an instruction group (*see* Shintani, col. 7, ll. 49-55), Shintani teaches that each of the pre-fetched instruction (which Shintani refers to as *p* continuous instructions) is buffered *in-order* in its own instruction register (which Shintani refers to as *p* instructions registers). *See* Shintani col. 7, l. 67 - col. 8, l. 4. In other words, each of Shintani's registers holds a single instruction, ***not a group of instructions*** as in independent claim 30. Thus,

Shintani does not teach or suggest "transferring the instruction group held in the pre-fetch buffer to a multiple-stage buffer when there is ***a vacancy*** in the multiple-stage buffer with sufficient capacity to handle the instruction group ***as a unit***," as in claim 30 (emphasis added). Accordingly, Shintani also does not teach or suggest "advancing ***an other instruction group held in the multiple-stage buffer*** so that another instruction from the other instruction group moves to the predetermined position," as in claim 30 (emphasis added).

The foregoing deficiencies of Shintani with respect to independent claim 30 are in no way remedied by the teachings of Hao. Consequently, independent claim 30 is patentable over the combination of Shintani and Hao. Dependent claims 31-35 are likewise patentable over the combination of Shintani and Hao for at least the same reasons as independent claim 30 from which they depend, and further in view of their own respective features. Accordingly, Applicants respectfully request that the rejection of claims 30-35 under 35 U.S.C. § 103(a) be reconsidered and withdrawn.

Independent claim 36 is directed to a data processing apparatus comprising a superscalar type microprocessor having a plurality of functional units that can execute instructions simultaneously. Among other features, the microprocessor of claim 36 includes "a buffer that buffers a plurality of instruction groups pre-fetched by the pre-fetch unit . . . wherein when a plurality of instructions of the instruction group are all retired, an entry in the buffer corresponding to the instruction group is released."

These features of independent claim 36 are not taught or suggested by Shintani. Shintani teaches a data processor that includes an instruction fetch circuit 201 that fetches instructions from a memory (1) into one of two prefetch instruction buffers 202

or 203. However, Shintani does not teach or suggest that instructions are stored in the buffers as "a plurality of instruction groups" as recited by independent claim 36. Furthermore, although Shintani does state that instructions are "extracted" from the instruction buffers 202 and 203 by instruction fetch circuit 201, Shintani nowhere teaches or suggests that when a plurality of instructions of an instruction group are retired, an entry in the buffer corresponding to the instruction group is released as recited by independent claim 36.

As far as Applicants understand the Examiner's position, the Examiner asserts that Shintani teaches "wherein when a plurality of instructions of the instruction group are all retired, an entry in the buffer corresponding to the instruction group is released," as recited in independent claim 36, because, according to the Examiner, "no specific type [of] retirement of the instructions has been reflected into the claims." (*See* the Office Action at p. 4, ¶ 12.) However, the Examiner's assertion is irrelevant. As set forth above, Shintani does not even teach or suggest a buffer that buffers a plurality of instruction groups. Thus, Shintani cannot possibly teach or suggest an entry in the buffer corresponding to the instruction group being released, as recited in claim 36.

The foregoing deficiencies of Shintani with respect to independent claim 36 are in no way remedied by the teachings of Hao. Consequently, independent claim 36 is patentable over the combination of Shintani and Hao. Dependent claim 37 is likewise patentable over the combination of Shintani and Hao for at least the same reasons as independent claim 36 from which it depends, and further in view of its own features. Accordingly, Applicants respectfully request that the rejection of claims 36 and 37 under 35 U.S.C. § 103(a) be reconsidered and withdrawn.

As currently amended, independent claim 38 recites a data processing apparatus comprising a super scalar type microprocessor having a plurality of functional units that can execute instructions simultaneously, the microprocessor comprising:

> a pre-fetch unit that pre-fetches a plurality of instructions from a memory in preparation for execution by one or more functional units, the plurality of instructions having a predetermined program order;
>
> a branch prediction circuit configured to provide a branch bias signal indicating whether a conditional branch controlled by a conditional branch instruction is predicted to be taken or not taken;
>
> a buffer that holds a plurality of instruction groups, including one or more instruction groups pre-fetched by the pre-fetch unit according to the branch bias signal;
>
> a decoder that simultaneously decodes a plurality of instructions from an instruction group held in the buffer;
>
> a register file including a plurality of registers used in the one or more functional units executing the plurality of decoded instructions;
>
> a dependency check unit that checks for a dependency relation between the plurality of instructions output from the decoder, on the basis of use conditions stored in a register;
>
> an instruction unit that allocates an instruction to a functional unit so that the instruction executes outside the predetermined program order after the instruction is judged by the dependency check unit not to be subject to restriction due to a dependency; and
>
> a retirement unit that specifies a register in which to store a result of executing the instruction outside the predetermined program order, wherein the retirement unit retires the instruction in program order after the instruction is completed.

Shintani and Hao, alone or in combination, do not teach or suggest each and every feature of independent claim 38. As set forth above, Shintani discloses *p* registers that buffer *p* instructions (one instruction per register), not a buffer that holds a plurality of groups of instructions. Hao also does not teach or suggest a buffer that holds a plurality of groups of instructions. Thus, neither Shintani nor Hao teach or suggest "a buffer that holds ***a plurality of instruction groups***," as recited in independent claim 38

(emphasis added). Furthermore, neither Shintani nor Hao teach or suggest, for example, a branch prediction circuit as in independent claim 38. Thus, Shintani and Hao, alone or in combination, cannot possibly teach or suggest "one or more instruction groups pre-fetched by the pre-fetch unit ***according to the branch bias signal***," as recited in independent claim 38 (emphasis added).

Because the combination of Shintani and Hao do not teach or suggest each and every feature of independent claim 38, this claim is patentable over these references. Dependent claims 39-40 are likewise patentable over the combination of Shintani and Hao for at least the same reasons as independent claim 38 from which they depend, and further in view of their own respective features. Accordingly, Applicants respectfully request that the rejection of claims 38-40 under 35 U.S.C. § 103(a) be reconsidered and withdrawn.

Independent claim 41 is directed to a data processing apparatus comprising a superscalar type microprocessor having a plurality of functional units that can execute instructions simultaneously. Among other features, the microprocessor of claim 41 includes "a buffer having plural stage registers that can transfer stored data in a forward direction and that stores the plurality of instructions of the pre-fetched instruction group" and "an instruction completion unit that advances contents of a plurality of registers of the buffer in the forward direction by a number of stages that correspond to a number of groups of completed instructions."

As set forth above, neither Shintani nor Hao teach or suggest a buffer that stores a plurality of groups of instructions. Thus, Shintani and Hao, alone or in combination, cannot possibly teach or suggest "an instruction completion unit that advances contents

of a plurality of registers of the buffer in the forward direction by a number of stages that correspond to a number of groups of completed instructions," as recited in independent claim 41 (emphasis added).

As far as Applicants understand the Examiner's position, the Examiner asserts that this claim feature is taught by Shintani because, in order for an instruction execution cycle to complete, the contents of the registers must be advanced in a forward direction by a number of pipeline stages — *i.e.*, a prefetch stage, a read stage, a decode stage, an execution stage, a data access stage, and a write stage. (*See* the Office Action at p. 4, ¶ 1.) However, claim 41 clearly recites that the contents of the buffer are advances "by a number of stages that ***correspond*** to a number of groups of completed instructions." The recited "number of groups of completed instructions" is not encompassed by a pipeline stage. Thus, the Examiner's position misapprehends the plain language of claim 41.

Because the combination of Shintani and Hao does not teach each and every feature of independent claim 41, this claim is patentable over these references. Dependent claim 42 is likewise patentable over the combination of Shintani and Hao for at least the same reasons as independent claim 41 from which it depends, and further in view of its own features. Accordingly, Applicants respectfully request that the rejection of claims 41 and 42 under 35 U.S.C. § 103(a) be reconsidered and withdrawn.

Conclusion

All of the stated grounds of objection and rejection have been properly traversed, accommodated, or rendered moot. Applicants therefore respectfully request that the Examiner reconsider all presently outstanding objections and rejections and that they be withdrawn. Applicants believe that a full and complete reply has been made to the

outstanding Office Action and, as such, the present application is in condition for allowance. If the Examiner believes, for any reason, that personal communication will expedite prosecution of this application, the Examiner is invited to telephone the undersigned at the number provided.

Prompt and favorable consideration of this Amendment and Reply is respectfully requested.

Respectfully submitted,

STERNE, KESSLER, GOLDSTEIN & FOX P.L.L.C.

Donald J. Featherstone
Attorney for Applicants
Registration No. 33,876

Date: 5/25/07

1100 New York Avenue, N.W.
Washington, D.C. 20005-3934
(202) 371-2600

675069_1.DOC